**1. Basic Queries**

* **SELECT**: Retrieve data from a table.

sql

SELECT column1, column2 FROM table\_name;

* **WHERE**: Filter data based on a condition.

sql

SELECT \* FROM table\_name WHERE column\_name = 'value';

* **ORDER BY**: Sort query results in ascending or descending order.

sql

SELECT \* FROM table\_name ORDER BY column\_name ASC; -- or DESC

**2. Aggregation Queries**

* **COUNT**: Count the number of rows.

sql

SELECT COUNT(\*) FROM table\_name;

* **SUM**: Calculate the total of a column.

sql

SELECT SUM(column\_name) FROM table\_name;

* **AVG**: Calculate the average value.

sql

SELECT AVG(column\_name) FROM table\_name;

* **MIN/MAX**: Find the smallest or largest value.

sql

SELECT MIN(column\_name), MAX(column\_name) FROM table\_name;

**3. Conditional Queries**

* **LIKE**: Search for patterns.

sql

SELECT \* FROM table\_name WHERE column\_name LIKE 'A%'; -- Starts with A

SELECT \* FROM table\_name WHERE column\_name LIKE '%A'; -- Ends with A

SELECT \* FROM table\_name WHERE column\_name LIKE '%A%'; -- Contains A

* **IN**: Check if a value is within a list.

sql

SELECT \* FROM table\_name WHERE column\_name IN ('value1', 'value2');

* **BETWEEN**: Filter values within a range.

sql

SELECT \* FROM table\_name WHERE column\_name BETWEEN value1 AND value2;

**4. Join Queries**

* **INNER JOIN**: Combine rows from two tables with matching values.

sql

SELECT a.column1, b.column2

FROM table\_a a

INNER JOIN table\_b b

ON a.common\_column = b.common\_column;

* **LEFT JOIN**: Return all rows from the left table and matched rows from the right table.

sql

SELECT a.column1, b.column2

FROM table\_a a

LEFT JOIN table\_b b

ON a.common\_column = b.common\_column;

* **RIGHT JOIN**: Return all rows from the right table and matched rows from the left table.

sql

SELECT a.column1, b.column2

FROM table\_a a

RIGHT JOIN table\_b b

ON a.common\_column = b.common\_column;

* **FULL OUTER JOIN**: Return rows when there is a match in either table.

sql

SELECT a.column1, b.column2

FROM table\_a a

FULL OUTER JOIN table\_b b

ON a.common\_column = b.common\_column;

**5. Data Modification Queries**

* **INSERT**: Add a new record to a table.

sql

INSERT INTO table\_name (column1, column2)

VALUES (value1, value2);

* **UPDATE**: Modify existing data in a table.

sql

UPDATE table\_name

SET column\_name = new\_value

WHERE condition;

* **DELETE**: Remove data from a table.

sql

DELETE FROM table\_name WHERE condition;

**6. Table Management Queries**

* **CREATE TABLE**: Create a new table.

sql

CREATE TABLE table\_name (

column1 datatype,

column2 datatype,

...

);

* **ALTER TABLE**: Modify an existing table.

sql

ALTER TABLE table\_name ADD column\_name datatype;

* **DROP TABLE**: Delete a table permanently.

sql

DROP TABLE table\_name;

**7. Advanced Queries**

* **GROUP BY**: Group rows sharing a property and aggregate data.

sql

SELECT column\_name, COUNT(\*)

FROM table\_name

GROUP BY column\_name;

* **HAVING**: Filter aggregated data.

sql

SELECT column\_name, COUNT(\*)

FROM table\_name

GROUP BY column\_name

HAVING COUNT(\*) > 5;

* **UNION**: Combine results from two queries (distinct values).

sql

SELECT column\_name FROM table\_a

UNION

SELECT column\_name FROM table\_b;

* **UNION ALL**: Combine results from two queries (including duplicates).

sql

SELECT column\_name FROM table\_a

UNION ALL

SELECT column\_name FROM table\_b;

**8. Subqueries**

* **Inline Subquery**: Use a query within another query.

sql

SELECT \* FROM table\_name

WHERE column\_name = (SELECT MAX(column\_name) FROM table\_name);

* **Correlated Subquery**: Use a query that references the outer query.

sql

SELECT a.column1

FROM table\_a a

WHERE column1 > (SELECT AVG(column2) FROM table\_b b WHERE a.id = b.id);